International Journal of Computer Science Engineering
and Information Technology Research (IJCSEITR)
ISSN(P): 2249-6831; ISSN(E): 2249-7943
Vol. 3, Issue 5, Dec 2013, 65-76

EFFICIENT INTRUSION CLASSIFICATION IN NETWORK USING SUPERVISED DATA MINING TECHNIQUES

SUSHIL KUMAR CHATURVEDI & ANAND JAWDEKAR

Department of CSE, SRCEM, Banmore, Gwalior, Madhya Pradesh, India

ABSTRACT

Security is a vast area of research. As we know there are three level of security which we are generally using. At first level we can assign user id and password, at second level file permission (read, write, execute) is used and at third level of defense we can use encryption and decryption. But these security levels provide static protection against intrusions. In this paper we are proposed a model for dynamic protection against intrusions.In this paper, well known intrusion dataset kddcup99 used for simulation purpose. This experimental model contains Information gain (IG) for dimension reduction, C4.5 for classifying attack classes and Random Forest used for optimization purpose..

KEYWORDS: Data Mining; C4.5; RF; IG, kddcup99

INTRODUCTION

Data is very vital to an organization. Administrations regularly desire to preserve the confidentiality of their audit record. The internet is used all over the world, it has become a key task to retain the privacy and veracity of the organization's audit record. The goal of attack detection is to build a system which would automatically scan network activity and detect such malicious attacks. Once an attack is detected, the system administrator could be informed and thus take counteractive action.Traditional security such as firewall, VPN and data encryption is insufficient to detect against attacks by crackers. However, attack finding is a activé one, which can give run time protection to the network security in monitoring, attack and counterattack [1]. For collecting the data set, the Attack Detection System (ADS) can be classified as host-based and network-based [2].

- **Host Based ADS:** -These types of systems actually run on the system being monitored. These data come from the records of different host system activities, including appraisal records of OS, system logs, application program statistics, and so on.
- **Network Based ADS:** -These types of system are placed on the network, near the computer system or system being observed. They examine the network stream of traffic and check whether it falls within acceptable boundaries. These data come through network segments, such as : IP packets.

Attack finding methods are categorized into two types [3]:

- **Anomaly Detection:** Anomaly detection refers to storing features of user's usual behaviors into a database, then comparing user's current behavior with those in the database. If the deviation is huge enough, we can say that there is something abnormal.
- **Misuse/Signature Detection:** Misuse Detection refers to confirming attack incidents by matching features through the attacking feature library.

We decided to use data mining in solving the problem of network attack because of following reasons [1, 4, 5, 6,]:

- Data mining methods can manage vast amounts of data.
- Data mining can easily find hidden information from vast amounts of data.

Data mining algorithms are used to perform data summarization and visualization that help the security analysis in several research areas [7].

RELATED WORK

All Denning was amongst the first persons to think in the area of application of data mining to internet security. He has proposed a model of a real –time intrusion-detection expert system [8]. Ming Xue give two main algorithms namely the pattern comparison and clustering algorithm . In pattern comparison, they first establish a normal behavior pattern under association rules and sequence rules then they distinguish normal behavior and intrusion behavior. The basic idea of clustering analysis originates in the difference between intrusion and normal pattern and in the fact that the number of normal patterns should exceed that of intrusion (attack) pattern, so that we can put data sets into different categories and detect intrusion by distinguishing normal and abnormal behaviors [1]. M. Govindarajan and RM. Chandrasekaran) investigated new techniques for intrusion detection model. They used comparative cross validation method for error rate for base classifiers. After this they discovered the general K-nearest neighbor (K-NN) classifiers as an intrusion detection model[9]. Mohammadreza Ektefa,Sara Memar,Fatimah Sidi and Lilly Suriani Affendey use C4.5 and SVM(support vector machine) for detecting attacks. They calculate the detection rate (percentage of detecting attacks among all attack data) and false alarm rate (percentage of normal data which is wrongly recognized as an attack) and compare both algorithm result and find C4.5 has better performance than SVM in both detection and false alarm rate[10].

PROPOSED METHODOLOGY

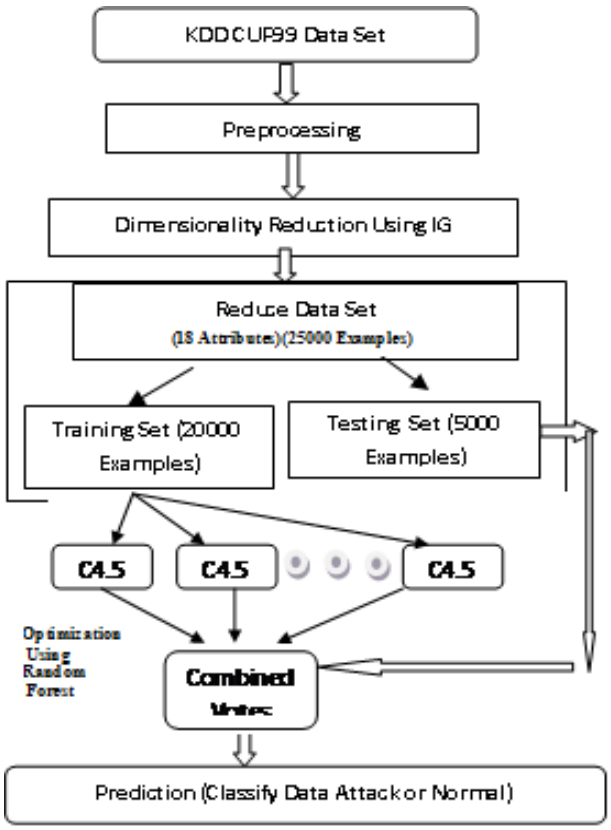

Figure 1: The Complete Proposed Model

In this paper attacks are detected using C4.5 classification algorithm then apply optimization technique such as Random Forest to improve the result of C4.5 algorithm. I have also used Information Gain (IG) to reduce the dimensionality of datasets required to detect attacks. The feature reduction process can be viewed as a preprocessing step which removes distracting variance from a data set, so that classifiers can perform better and gives better results. In our proposed algorithm, Information Gain (IG) transforms used for dimensionality reduction which is commonly used step, especially when dealing with the high dimensional space of features. IG-based approaches improve system performances. In this thesis comparison of existing algorithms (Naive Bayes and C4.5) with proposed Classifier is used to classify any unknown attacks and result of the proposed algorithms is compared with Naive Bayes and C4.5 to show which one is the best algorithm to classify new kind of attacks.

The steps perform in this proposed model is given below:

- **Data Preprocessing** –Data preprocessing comprises following components including document conversion and feature allowance. The working of each module is described as follows:
    - Dataset selection having DoS, r2l, u2r and Probe attacks.
    - Dataset translation- converts different types of data such as gz, Tcpdump to csv file and arff (Attribute-Relation File Format) data file format.
- **Dimensionality Reduction (Feature Selection**) – It reduces the dimensionality of the data space by removing irrelevant or less relevant feature selection criterion. Information Gain (IG) is used for dimensionality reduction. The goal of IG is to reduce the dimensionality of the data while retaining as much as possible of the variation present in the original dataset.
- **Classification Using C4.5 Decision Tree Algorithm** - Decision Trees (DT) learning algorithms work based on processing and deciding upon nodes of the data. Nodes in DT are attributes and each leaf node is signifying a classification. C4.5 algorithm steps are used to create a decision tree developed by Ross Quinlan. C4.5 is an edition of Quinlan's earlier ID3 algorithm. The decision tree created by C4.5 can be used for prediction of class level, and for this reason, C4.5 is often denoted as a statistical classifier.C4.5 uses information entropy concept [11]. C4.5 builds decision trees from a set of training data in the similar manner as ID3, using the concept of information entropy.
- **Optimization through Random Forest**: The random forests [13] is an ensemble of unpruned classification or RT(regression trees). Random forest creates various classification trees. Each tree is built from a different bootstrap sample from the original data using a supervised algorithm (Like C4.5). After the forest is made, a new record that needs to be classified is put down each of the generated trees in the forest for prediction of attack type. Each tree gives a decision that indicates the tree's result about the class of the record. This method picks the class for which most decisions for the record [12].

CLASSIFIER FOR ATTACK DETECTION

Information Gain (IG)

Information gain is a reasonable method to use for feature selection (even when there are many classes exist on dataset). The information gain(IG) is a method for selecting the decision attributes for constructing decision trees. Note that a typical problem with tree generation is when to stop adding decision nodes too many nodes usually leads to poor

generality. This method will help you decide an collection of features from the most suitable to least suitable. To determine cutoff point another method (such as evaluation on a holdout set) is used [16].

Algorithm

Step:-1 Calculate expected information needed to classify a given sample is given by

$$I(d_1, d_2, ... d_m) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

Where d_i be the number of samples of D in class C_i .

$$p_i = \frac{d_i}{D}$$

p_i is the probability that an attribute sample belongs to class C_i

Step 2:- Calculate the entropy or expected information based on the partitioning into subsets by attribute A.

$$E(A) = -\sum_{j=1}^{v} |D_v|/|D| I(d_{1j}, d_{2j}, ..., d_{mj})$$

Where $|D_v| = (d_{1j} + d_{2j} + ...+ d_{mj})$

| D | = Total no of samples

Where $p_{ij} = d_{ij} / |D_j|$ is the probability that a sample in D_j belongs to class C_i .

Step 3:- calculate encoding information that would be gained by branching on A is

Gain (A) = I (d1, d2, ……., d_m) – E(A)

Gain (A) is the estimated minimization in entropy caused by knowing the value of Attribute A.

Step 4:- Repeat step 2 to 3 and Compute the information gain of each attribute.

Step 5:- Sort these information gain values and select attributes having large information gain.

After applying information gain selected attributes are given below:

18 attributes are relevant out of 42 attributes for attack detection. Selected attribute names are service, src_bytes, count, dst_bytes, srv_count, dst_host_srv_count, dst_host_diff_srv_rate, dst_host_same_src_port_rate, flag,diff_srv_rate, dst_host_same_srv_rate, logged_in, same_srv_rate, dst_host_serror_rate, serror_rate, protocol_type, dst_host_rerror_rate, dst_host_count.

Naive Bayesian Classifier(NBC)

Bayesian classifiers are statistical classifiers. They can calculate class membership possibilities, such as the possibility that a given record belongs to a particular class. Bayesian classification is built on Bayes theorem. Comparison of different classification algorithms have originate a simple Bayesian classifier known as the naive Bayesian classifier to be comparable in performance with decision tree and neural network classification algorithm. Bayesian classifiers have also revealed high accuracy and speed when applied to the huge amount of datasets. [17].

Naive Bayesian classifiers assume that the effect of an attribute value of a given class is independent of the values of the other attributes. This hypothesis is called class conditional independence. It is made to simplify the calculations involved and, in this sense, is considered "naive".

Algorithm steps are as follows:-

Input: - Training data samples having attribute set $A=\{A_1,A_2,\ldots\ldots,A_n\}$,Unknown data sample $X=\{x_1,x_2,\ldots\ldots,x_n\}$, suppose there are m classes $C=\{C_1, C_2,\ldots\ldots,C_m\}$

Output: - Predict the class label of the unknown data sample X.

Step1: calculate the prior probability of the class label

$$P(C_i)=\frac{S_i}{S}$$

Where Si is the total of training samples of class Ci , and S is the total number of training samples.

Step2: Calculate the probability of unknown data sample with respect to class label.

$$P(X \mid C_i)=\prod_{k=1}^{n} P(x_k \mid C_i)$$

The probability $P(x_1 \mid C_i), P(x_1 \mid C_i), \ldots\ldots\ldots, P(x_n \mid C_i)$ can be estimated from the training samples.

If A_k is categorical then, $P(x_k \mid C_i)=\frac{S_{ik}}{S_i}$ where S_{ik} is the number of training samples of class C_i having the value x_k for A_k , and S_i is the number of training samples belonging to C_i .

If A_k is continuous valued, then the attribute is typically assumed to have a Gaussian distribution so that

$$P(x_k \mid C_i)=g(x_{k,}\mu c_i,\sigma c_i)=\frac{1}{\sqrt{2\pi\sigma c_i}}e^{\frac{(x_k-\mu c_i)^2}{2\sigma c_i^{\,2}}}$$

Where $g(x_k,\mu c_i,\sigma c_i)$ is the Gaussian(normal) density function for attribute A_k , while μc_i and σc_i are the mean and standard deviation, respectively,given the values for attribute A_k for training samples of class C_i .

Step3:- compute $P(X \mid C_i)P(C_i)$ for each class C_i.

Step4:- Sample X is be appropriate to the class C_i if and only if $P(X \mid C_i)P(C_i) > P(X \mid C_j)P(C_j)$ for $1 \le j \le m, j \ne i,$ or in another way it is assigned to class C_i for which $P(X \mid C_i)P(C_i)$ is the maximum.

C4.5(Decision Tree)

C4.5 is a suite of algorithms for classification problems in machine learning and data analysis. It is belong to supervised learning: Given an attribute value data set where instances are described by collections of attributes and belong to one of a set of mutually exclusive classes, C4.5 studies a mapping from attribute values to classes that can be applied to classify new, unseen instances.[21] All tree induction methods begin with a root node that represents the entire, given data set and recursively split the data into smaller subsets by testing for a given attribute of each node. The sub trees denote the partitions of the original dataset that fulfill specified attribute value tests. This method process is usually continued until the subsets are ''pure'' that is, all illustrations in the subset fall in the similar class, at which time the tree growing is terminated [18]. This algorithm can be used to generate a decision tree that can be used to classify data instances in

different classes which helps in further analysis detecting valid results. This algorithm made a number of improvements on ID3 supervised algorithm. These are given below:

- It can handle both continuous and discrete attributes. For continuous attributes, it creates a threshold and then splits the list into those whose attribute value is above the threshold and those that are less than or equal to it.
- The attribute has no values are simply not used in gain and entropy calculations.
- It can handle attributes with different costs.

Algorithm (C4.5) process steps are as follows:

Input: an attribute-valued dataset D (after applying Dimensionality reduction method IG)

1. Tree = {}
2. if D is "pure" OR other stopping criteria met then
3. terminate
4. end if
5. for all attribute a ϵ D do
6. Calculate information-theoretic criteria if we splitting on a
7. end for
8. a_{best} = Best attribute according to above computed criteria
9. Tree = Create a decision node that tests a_{best} in the root
10. D_v = Induced sub-datasets from D based on a_{best}
11. for all D_v do
12. $Tree_v$ = C4.5(D_v)
13. Append $Tree_v$ to the corresponding branch of Tree
14. end for
15. return Tree

Random Forest(RF)

The following are the features of random forests algorithm:

- It is unsurpassable in accuracy among the existing data mining algorithms.
- It runs proficiently on large volume data sets with many features.
- It can give the approximations of what features are important.
- It has no insignificant data problem and does not over-fit.
- It can handle unbalanced data sets.

In random forests, there is no need for cross-validation or a test set to get an unbiased estimate of the test error. There are two ways to evaluate the error rate. One is to divide the dataset into a training set and test set. We can employ the training set to form the forest, and then use the test set to compute the error rate. In the second way, use the oob error estimate. Because RF algorithm computes the oob error throughout the training stage, we do not need to divide the training data.in this paper, select the oob error estimate, since it is more operative by learning from the whole training dataset [13]. Fig.2 below is a visual representation of the un-weighted random forest algorithm.

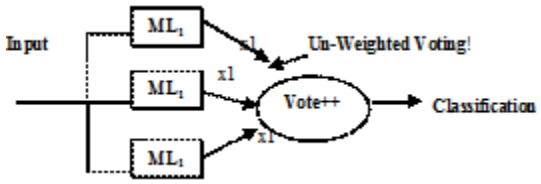

Figure 2: Meta Learner [14]

Random Forest(RF) algorithm procedure is as follows:Build bootstrapped sample B_i from the original dataset D, where $|B_i| = |D|$ and examples are chosen at random with replacement from D.

- Construct a tree T_i using B_i as the training dataset using the standard decision tree algorithm(C4.5) with the following modifications:
    - At each node in the tree , restrict the set of candidate attributes to a randomly selected subset (x_1, x_2, x_3, … , x_k), where k = no. of features.
    - Do not prune the tree.
- Repeat steps (1) and (2) for i = 1, … , no. of trees, creating a forest of trees T_i derived from different bootstrap samples.

When classifying an example x, aggregate the decisions (votes) over all trees T_i in the forest. If T_i (x) is the class of x as determined by tree T_i , then the predicted class of x is the class that occurs most often in the ensemble, i.e. the class with the majority votes[15].

EXPERIMENTS

In this paper, our work is tested using the 1999 KDD cup network anomaly data set [19]. The first stage is pre-processing. The data in this phase are reduced to lower dimensionality (18 attributes) then partition into training and testing. In the next step, we applied C4.5 and NBC on the training dataset in order to build and train the models.After training, trained model is estimated on the testing dataset to calculate the efficiency of the model.The training data set consists of seven weeks of traffic with around 5 million connections and the testing data consists of two weeks of traffic with around 300,000 connections. The data contains four main categories of attacks (1)Denial-of-service (DoS) such as land, neptune, back, etc.(2) Remote-to-local (R2L) like guesspasswd,warezclient, etc. (3) User to root (U2R) such as loadmodule, buffer_overflow and so on.

PROBING such as satan, insweep, portsweep, etc.

Mining algorithms can lead to improved results if data under analysis have been normalized [20].

Finding of attack can be dignified by following metrics:

- **False Positive (FP):** Or false alarm, Corresponds to the number of detected attacks but it is in fact normal.

- •**False Negative (FN):** Corresponds to the number of detected normal instances but it is actually attacked, in other words these attacks are the target of intrusion detection systems.
- **True Positive (TP):** Corresponds to the number of detected attacks and it is in fact attack.
- **True Negative (TN):** Relates to the number of detected normal instances and it is actually normal.

The accuracy of an intrusion detection system is measured regarding to detection rate and false alarm rate. In this work, we use 1999 KDD cup Dataset which consist of (25000 records). Table 1 given below shows the percentage of the data. Then, 15% of the data is extracted by sampling. 80% ofthis new set belonged to training set, and 20% dedicated to test data.

Table 1: Percentage of Attack Data

Attack Name	Training Data(20,000)	Testing Data(5,000)
	Quantity (Attack)	Percentage (Attack)
Normal	13,119	65.59
Dos	4,583	22.91
U2r	40	0.2
Probe	1820	9.1

Detection Rate Comparison

Detection rate refers to the percentage of detected attacks among all attack data, and is represented as follows:

$$DetectionRate = \frac{Detected\_Attack \times 100}{All\_Attack\_Data}$$

Or $$DetectionRate = \frac{TP \times 100}{TP + TN}$$

The results of detection rate for different types of attacks are shown in Table 2. As statistical outcomes indicate, the average detection rate for NBC (Naïve Bayes Classifier), C4.5 and Proposed algorithm(best result with 18 attributes) are 81.52, 89.72 and 96.00 respectively. Furthermore, the detection rate for proposed algorithm is better than NBC and C4.5. Table 2 Detection Rate Comparison of Different Attacks Through C4.5 ,NBC and Proposed Algorithm

Table 2: Comparison of Different Methods for Various Datasets

Algorithm/ Attack Type	7 Attribute Set			
	Dos	U2R	Probe	R2L
NBC	85.43	75.00	84.57	49.57
C4.5	65.86	35.00	93.11	40.33
Proposed	66.04	75.00	98.07	40.33
Algorithm/ Attack Type	**18 Attribute Set**			
	Dos	**U2R**	**Probe**	**R2L**
NBC	92.49	70.00	97.24	66.38
C4.5	97.40	85.00	97.52	78.99
Proposed	99.64	90.00	99.44	94.95
Algorithm/ Attack Type	**41 Attribute Set**			
	Dos	**U2R**	**Probe**	**R2L**
NBC	94.28	70.00	93.66	51.26
C4.5	96.42	70.00	96.96	40.33
Proposed	98.21	75.00	99.99	41.17

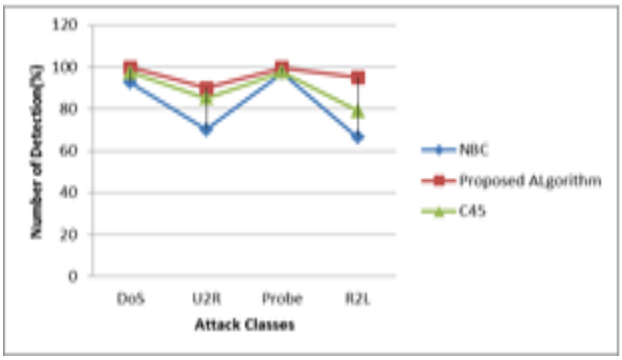

Figure 3: Comparison of the Proposed Algorithm with NBC&C45 Classifier

The proposed model applied for the different attribute sets, 18 attribute set gives better results as compare with other attributes set.

Comparison of Other Parameters

False alarm rate refers to the percentage of normal data which is wrongly recognized as an attack, and is represented as follows:

$$FalseAlarmRate = \frac{FP \times 100}{FP + FN}$$

The false positive rate in our experiment is 0.013 for NBC, 0.013 for C4.5 and 0.001 for Proposed Algorithm. As the results show, proposed algorithm also performs better in false positive rate than NBC and C4.5.

Table 3: Comparisons of other Parameters

Test Set	Parameter	Classifier		
		NBC	C45	Proposed Algorithm
41 Attribute Set	Accuracy(ACC)	95.58	96.44	97.84
	Detection Rate(DR)	90.68	92.10	94.75
	False Positive Rate(FPR)	0.020	0.014	0.006
	Recall	0.90	0.92	0.94
	Precision	0.95	0.96	0.98
	F-Measure	0.93	0.94	0.96
18 Attribute Set	Accuracy(ACC)	96.30	97.79	99.62
	Detection Rate(DR)	91.36	95.92	99.13
	False Positive Rate(FPR)	0.013	0.013	0.001
	Recall	0.91	0.95	0.99
	Precision	0.97	0.97	0.99
	F-Measure	0.94	0.96	0.99
7 Attribute Set	Accuracy(ACC)	92.00	89.14	90.10
	Detection Rate(DR)	82.47	69.71	71.43
	False Positive Rate(FPR)	0.034	0.015	0.009
	Recall	0.82	0.69	0.71
	Precision	0.92	0.95	0.97
	F-Measure	0.86	0.80	0.82

Here different parameters are calculated for various attributes set. Some values are improved such as detection rate, accuracy, precision, f-measure, recall and some values are minimized such as false rate with 18 attributes set.

Table 4: Comparison of Result with Other Methods

Algorithm/Attack Type	ACC	FPR	TPR
NB [16]	92.69	0.077	0.93
Bayes Net[16]	99.10	0.001	0.98
OneR[16]	92.69	0.002	0.96
NBC	96.30	0.013	0.91
C4.5	97.79	0.013	0.95
Proposed Algorithm	99.62	0.001	0.99

CONCLUSIONS

In this paper data mining techniques namely C4.5 and NBC and RF are used to detect anomaly in the network. Experiment results show, proposed algorithm has better results than NBC and C4.5 in both detection and false alarm rate in our data set.There are some challenges handled by the IDS. Like other supervised learning methods, the new type of attack cannot be easily discovered by these IDS. If new attack is found in the testing data, it is detected as a normal data however, users' behaviors change as time spends. After some time training data might become outdated and incomplete for prediction. Thus, it is advised a periodic updating to the training sets and profiles. These steps could be done off-line without affecting the on- line detection system. The accuracy of classification is not 100 percent. Many times normal processes get declared as malicious

To improve the detection capacity of the IDS, the future enhancement can be done as follows: the algorithm that has features of supervised and unsupervised technique will be developed.by using a supervised technique known type of attacks will be discovered and by using unsupervised technique unknown type or new type attacks will be detected. For dimensionality reduction a hybridized Rough-PCA Approach of Attribute Reduction will be used so high dimensionality of the data set is reduced to lower dimension with most relevant attribute set.

REFERENCES

1. M. Xue, C. Zhu. (2009) "Applied Research on Data Mining Algorithm in Network Intrusion Detection," jcai, pp. 275-277, 2009 International Joint Conference on Artificial Intelligence.
2. D. E. Denning. (1987) "An intrusion detection model," IEEE Transaction on Software Engineering.
3. T. Bhavani et al.(2008)"Data Mining for Security Applications," Proceedings of the 2008 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing - Volume 02, IEEE Computer Society.
4. T. Lappas and K. P. (2007) Data Mining Techniques for (Network) Intrusion Detection System.
5. S. Sun, Y. Wang. (2009) "A Weighted Support Vector Clustering Algorithm and its Application in Network Intrusion Detection," etcs, vol. 1, pp. 352-355, 2009 First International Workshop on Education Technology and Computer Science.
6. S. Wu, E. Yen. (2009). "Data mining-based intrusion detectors," Elsevier Computer Network.
7. E. Bloedorn et al.(2001) "Data Mining for Network Intrusion Detection: How to Get Started," Technical paper.
8. Dorothy E. Denning.(1986) "An Intrusion-Detection Model" IEEE Computer Society Symposium on Research in Security and Privacy , pp 118-31
9. M.Govindarajan and RM. Chandrasekaran. (2009) "Intrusion Detection using K-Nearest Neighbor" ICAC 2009 978-1-4244-4787-9/09 IEEE

10. Mohammadreza Ektefa , Sara Memar, Fatimah Sidi ,Lilly Suriani Affendey. (2010) “Intrusion Detection Using Data Mining Techniques” 978-1-4244-5651-2/10 IEEE.
11. Michael D. Alder.(1997) “ An Introduction to Pattern Recognition: Statistical,Neural Net and Synthetic Methods of getting robots to see and hear”, September19,http://ciips.ee.uwa.edu.au/mike/PatRec
12. J. Zhang, and M. Zulkernine,(2006). A Hybrid Network Intrusion Detection Technique Using Random Forests.In Proceedings of the IEEE First International Conference on Availability, Reliability and Security(ARES’06).
13. L. Breiman (2009), “Random Forests”, Machine Learning 45(1):5–32
14. White, Mark. (2005) ECE591Q-Machine Learning – Lecture slides, Fall.
15. T.M. Khoshgoftaar, M. Golawala and J. Van Hulse. (2007), “An Empirical Study of Learning from Imbalanced Data Using Random Forest." Proceedings of the 19th. IEEE Conference on Tools with Artificial Intelligence. 2007, pp. 310-317.
16. Yogendra Kumar jain and Upendra.(2012) “An Efficient Intrusion Detection Based on Decision Tree Classifier Using Feature Reduction ” International Journal of Scientific and Research Publications, ISSN 2250-3153,Volume 2,Issue 1.
17. J. Han, and M. Kamber. (2006), “Data mining: concepts and techniques’”(2nd ed.). Morgan Kaufmann Publishers.
18. Naren Ramakrishnan (2009). “C4.5” © 2009 by Taylor & Francis Group, LLC.
19. http://kdd.ics.uci.edu/databases/kddcup99/
20. J. Han, and M. Kamber.(2006), “Data mining: concepts and techniques’” (2nd ed.). Morgan Kaufmann Publishers.
21. Prabhjeet Kaur , Amit Kumar Sharma, Sudesh Kumar Prajapat (2012) “MADAM ID FOR INTRUSION DETECTION USING DATA MINING” IJRIM Volume 2, Issue 2 (ISSN 2231-4334).